QCD@LHC Workshop 2019

Application of the APPLfast grid interface to NNLOJET and determination of α_s at NNLO from DIS jet cross sections

D. Britzger (MPI Munich), C. Gwenlan (Oxford),
A. Huss (CERN), **K. Rabbertz (KIT)**, M. Sutton (Sussex)

Karlsruhe Institute of Technology

Motivation

- Interpretation of experimental data requires reasonably fast theory
- Often: Repeated computation of same cross section:
  - Different PDF sets; PDF uncertainties
  - Variations of renormalisation & factorisation scales μ_R, μ_F
  - Variation of $\alpha_s(M_Z)$
  - SM parameter fits (→ e.g. with xFitter, see talk by C. Gwenlan)
- Jet cross section calculations at NLO were slow
  - Initial reason for interpolation grids
- Nowadays NNLO in general very demanding!
  - More than ever need fast re-evaluations of higher order cross sections for varied input parameters
  - Use interpolation grids like from APPLgrid or fastNLO

APPLgrid, Carli et al., Eur. Phys. J. C, 2010, 66, 503.
fastNLO, Britzger et al., arXiv:1208.3641.

What is APPLfast?

- **Started as joint project of NNLOJET, APPLgrid, and fastNLO authors at QCD@LHC in London, 2015**

- **Developed common interface between NNLOJET and fast interpolation grid technology**

  - **The least obtrusive as possible for NNLOJET**
    - **No grids → no timing penalty**

  - **As similar as possible for APPLgrid & fastNLO**
    - **Simpler maintenance**
    - **Usable from other theory programs**

NNLOJET

X. Chen, J. Cruz-Martinez, J. Currie, R. Gauld, A. Gehrmann-De Ridder, T. Gehrmann , E.W.N. Glover, A. Huss, I. Majer, T. Morgan, J. Niehuis, J. Pires, D. Walker

Common framework for NNLO corrections using antenna subtraction

Characterictics:

- **Parton-level event generator**
- **Based on antenna subtraction**
- **Test & validation framework**
- **Interface to APPLfast**

Processes:

- $\mathrm{pp} \to (\mathrm{Z} \to \ell^+\ell^-) \ + 0, 1$ jets
- $\mathrm{pp} \to (\mathrm{W}^\pm \to \ell\nu) \ + 0, 1$ jets
- $\mathrm{pp} \to \mathrm{H} \ + 0, 1$ jets, VBF
  $\hookrightarrow \gamma\gamma, \ \ell^+\ell^-\gamma, \ 4\ell, \ldots$
- $\mathrm{pp} \to$ dijets
- $\mathrm{ep} \to 1, \ 2$ jets
- $\mathrm{e^+e^-} \to 3$ jets
- $\ldots$

A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, A. Huss, T. Morgan, PoS RADCOR2015 (2016) 075, arXiv: 1601.04569.

Interpolation concept

Implemented in APPLgrid & fastNLO

Use interpolation kernel

- Introduce set of n discrete **x-nodes**, x_i's being equidistant in a function f(x)
- Take set of **Eigenfunctions $E_i(x)$** around nodes x_i

→ Interpolation kernels

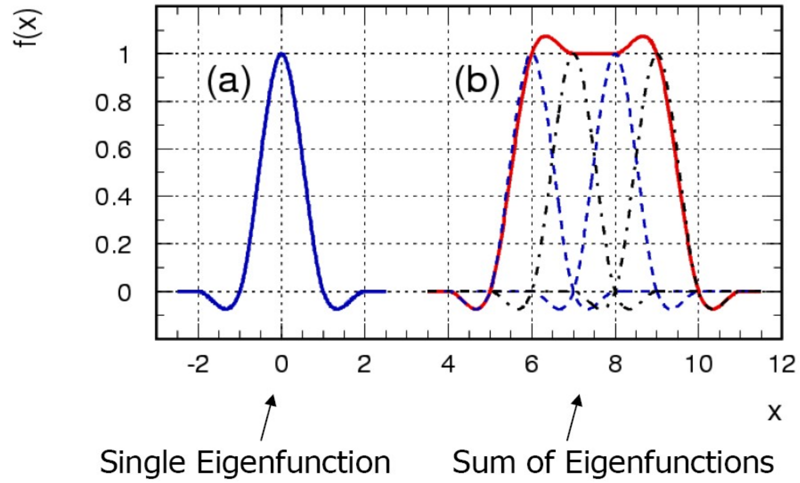
- Actually a rather old idea, see e.g. C. Pascaud, F. Zomer (Orsay, LAL), LAL-94-42

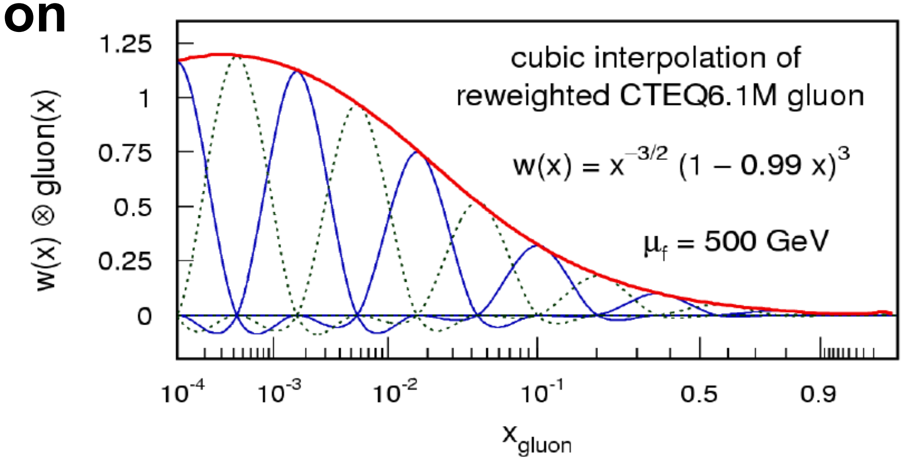
→ Single PDF is replaced by a linear combination of interpolation kernels

$$f_a(x) \cong \sum_i f_a(x_i) \cdot E^{(i)}(x)$$

→ Then the integrals are done only once

→ Afterwards only summation required to change PDF





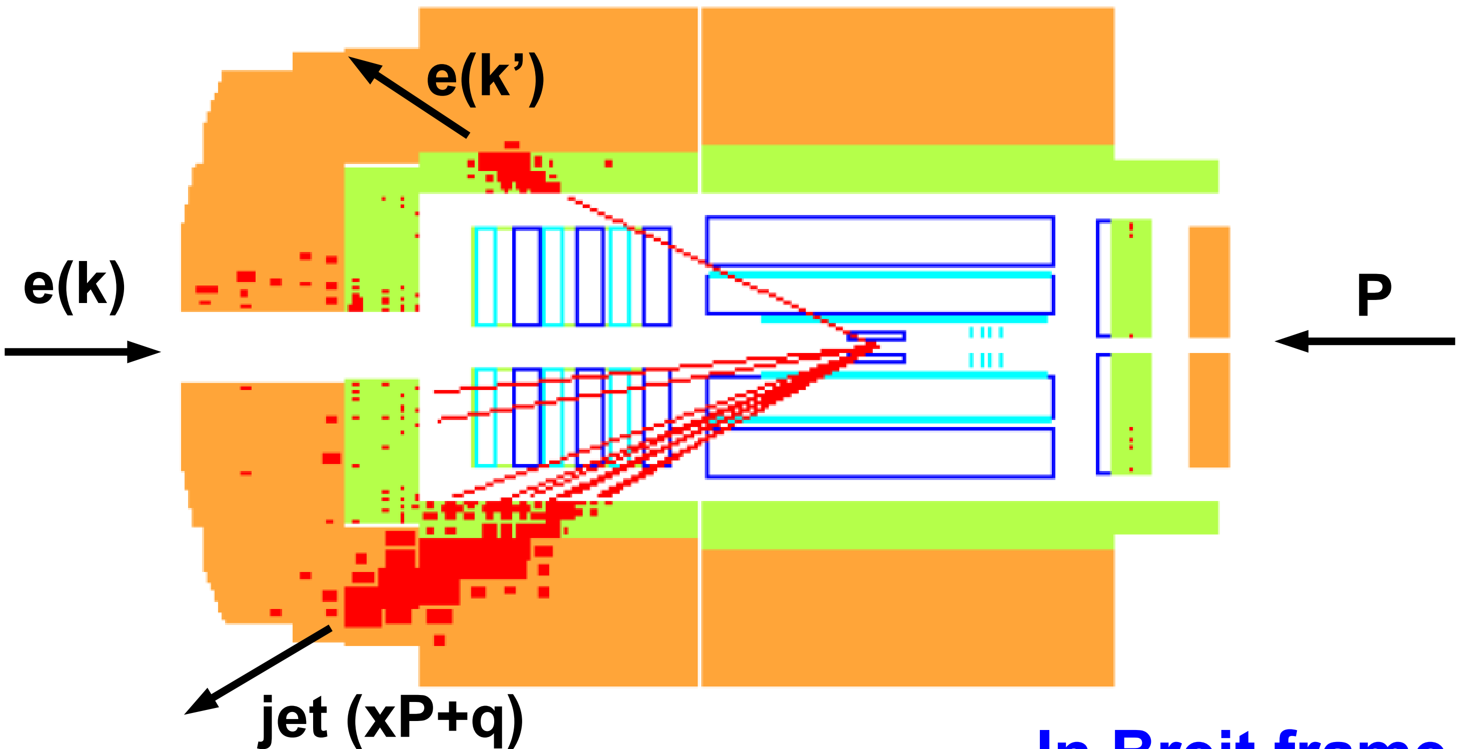
Tabulate the convolution of the perturbative coefficients with the interpolation kernel

NNLOJET+APPLfast workflow

- **1. Preprocessing:** Check of interpolation quality
  - Short test jobs to check interpolation settings (& optimise if necessary) O(10 h)
- **2. NNLOJET Warm-up:** Vegas integration optimisation
  - 1 long (multi-core) job per process O(100 h)
- **3. APPLgrid/fastNLO Warm-up:** Adapt x- and scale-grids to accessed phase space (exact strategy differs between APPLgrid & fastNLO)
  - Only phase space provided from NNLOJET → significant speed-up O(100 h)
- **4. Interpolation grid production:**
  - Thousands of parallel jobs O(250 kh)
- **5. Postprocessing:** Statistical evaluation and combination of all produced grids …
  - Job to combine all grids and estimate statistical uncertainty O(100 h)
- **6. Validate, validate, and validate** O(? h)
- **7. Present final results :-)** 30 min :-)

Application to DIS jets

DIS event in H1 detector



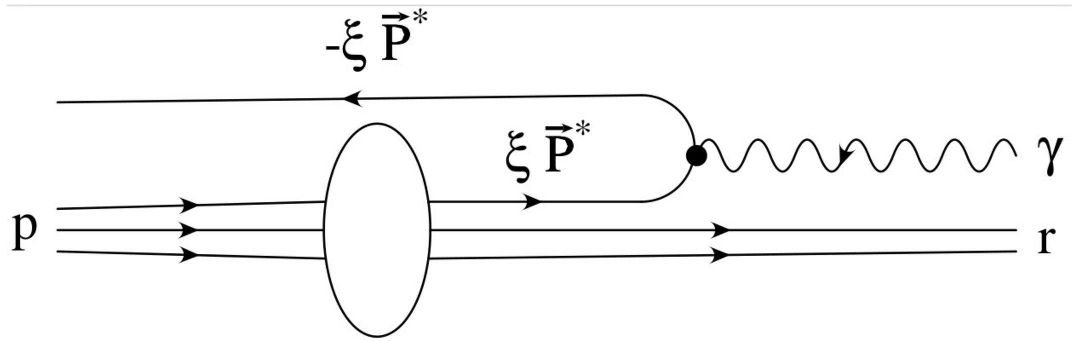
DIS kinematics

$Q^2 = -q^2$
$= -(k-k')^2$

$x = Q^2 / 2P \bullet q$

In Breit frame of reference

Backscattered quark in quark-parton-model

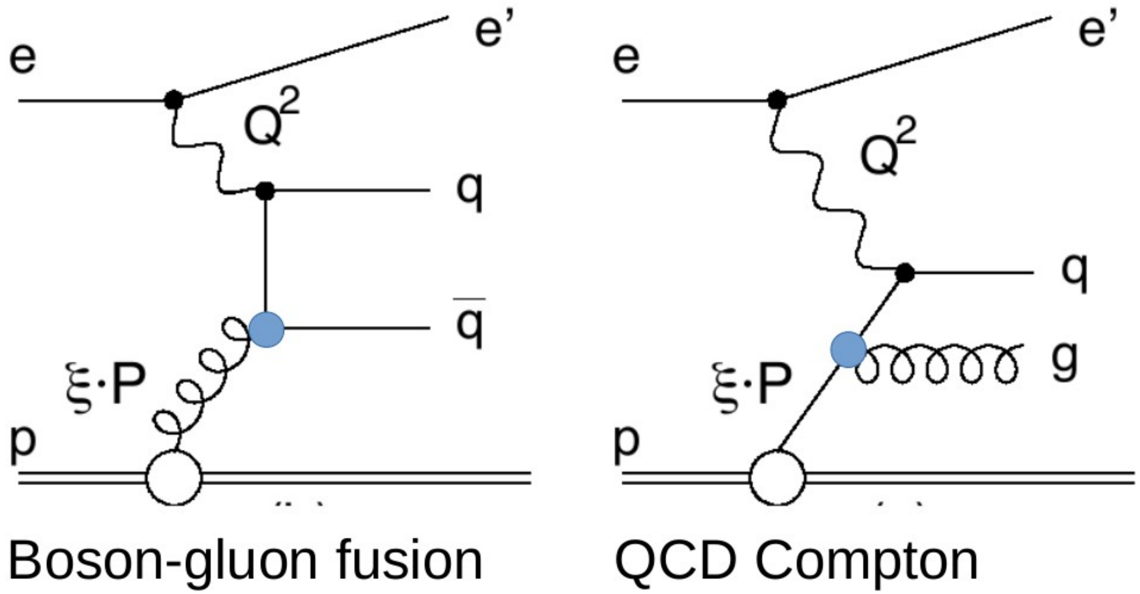


No jet pT!

DIS jets in Breit frame

ep → 2 jets

from head-on collision of virtual boson with struck quark in proton



Accesses gluon density → dominates at low to medium x

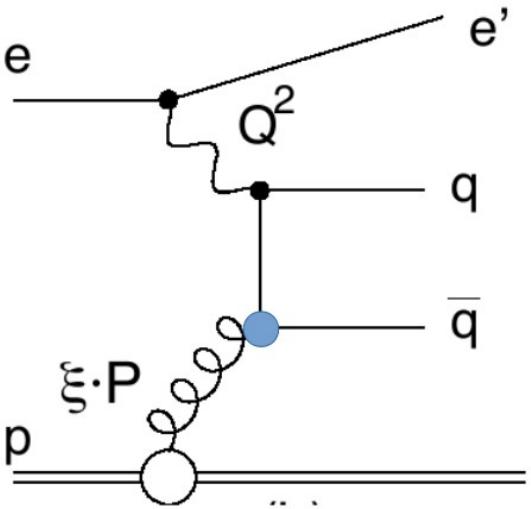
Accesses quark density → dominates at high pT (high x)

Sensitivity to strong coupling constant α_s and gluon density $g(x,\mu_F^2)$ at LO

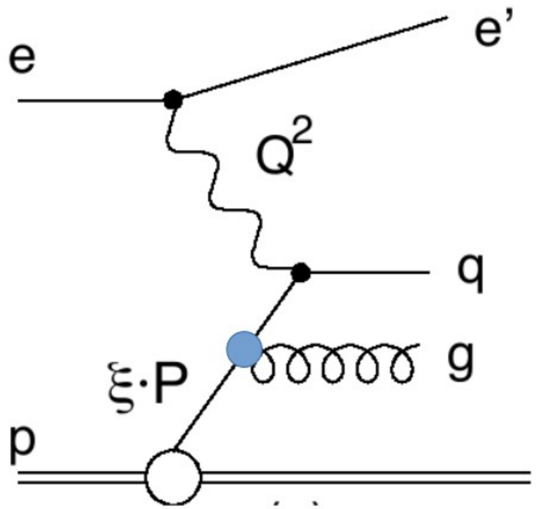
DIS jets in Breit frame

ep → 2 jets

from head-on collision of virtual boson with struck quark in proton



Boson-gluon fusion

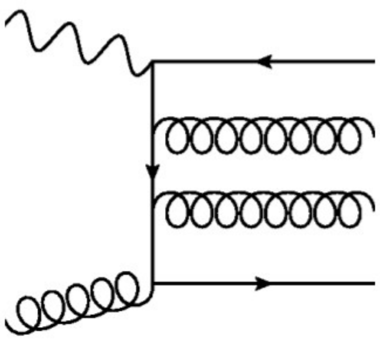


QCD Compton

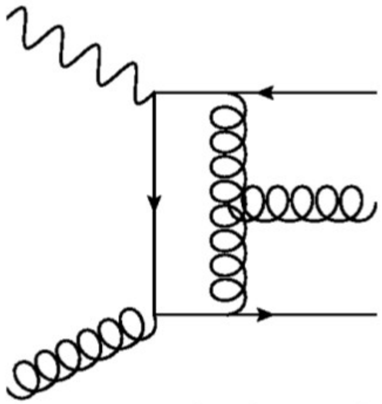
Accesses gluon density
→ dominates at low to medium x

Accesses quark density
→ dominates at high pT (high x)

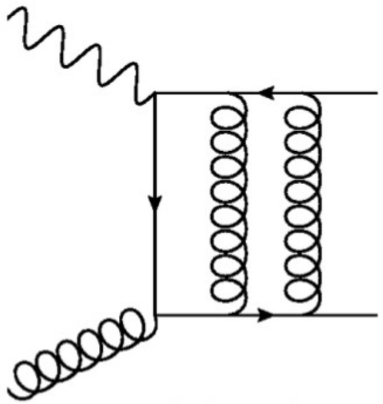
Now with NNLO corrections!



Double-real



Real-virtual



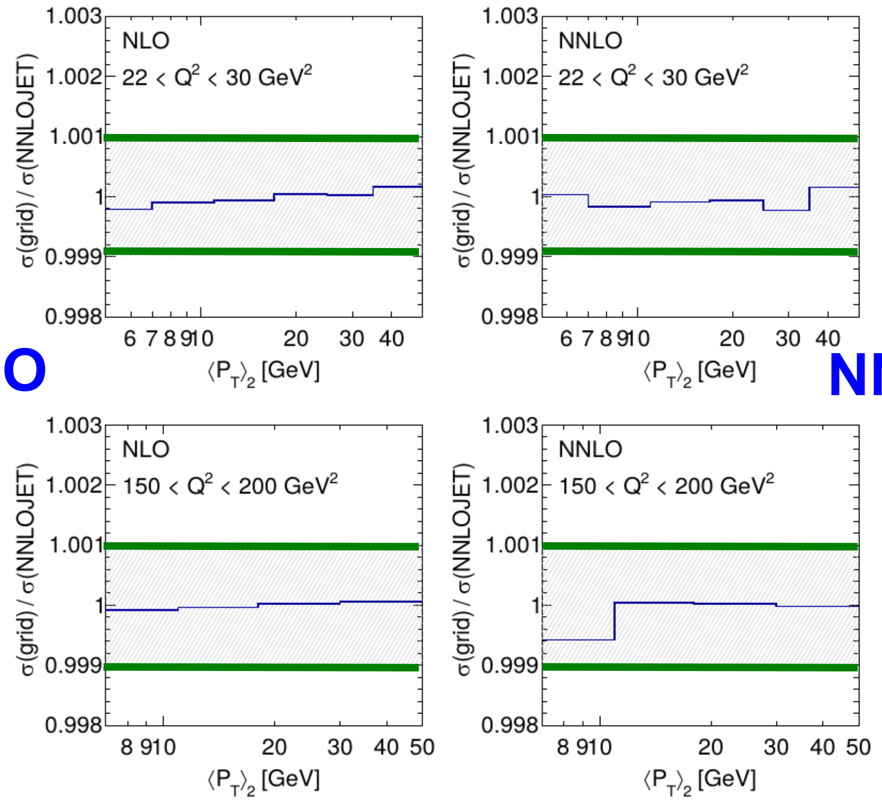
Double-virtual

Closure in x for DIS jet grids

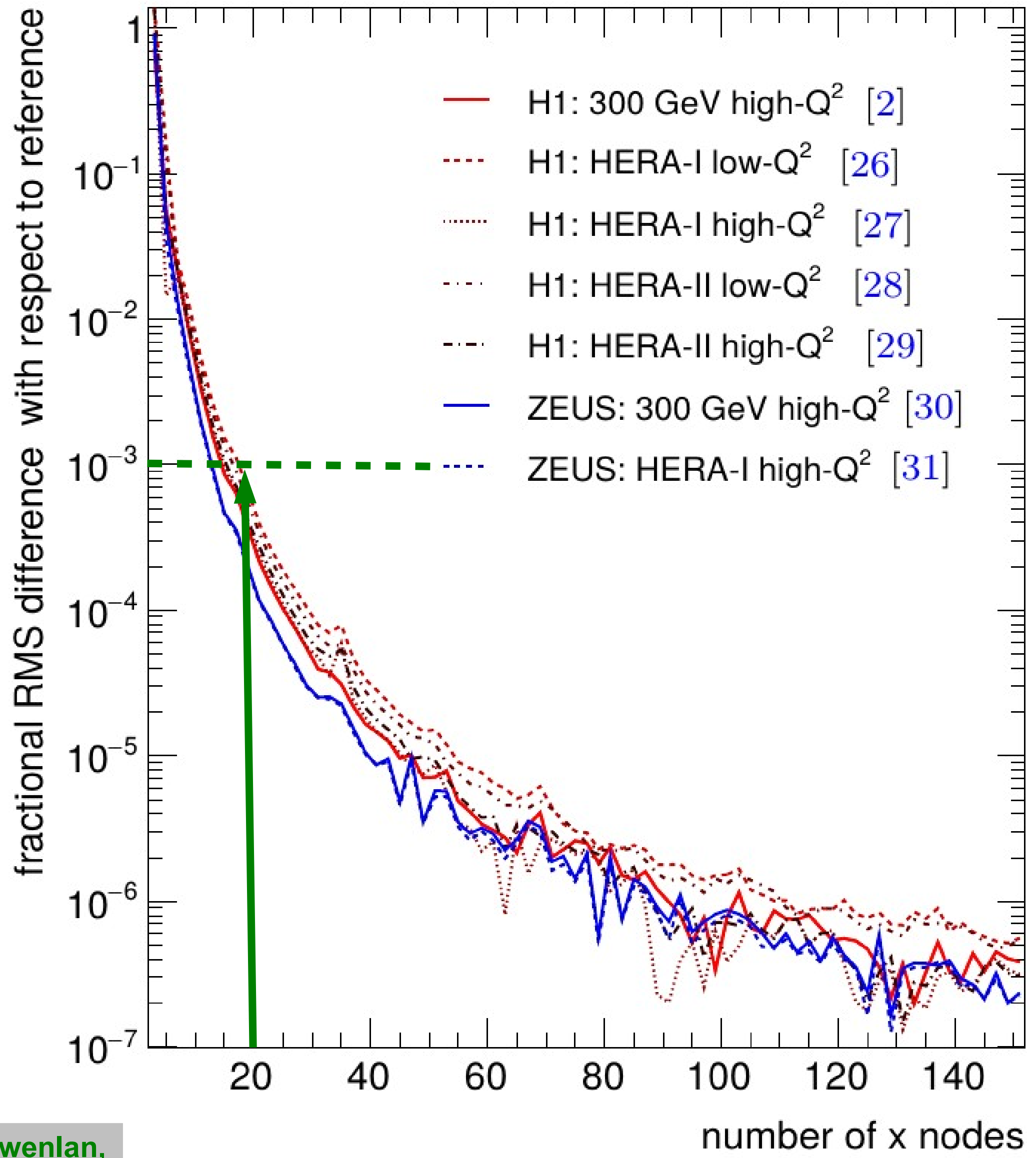
Interpolation quality optimisable, e.g. depends on process(!) and :

- **Reweighting functions**
- **Distribution of x nodes**
- **Number of x nodes**

NLO **NNLO**



With 20 nodes < 1 ‰ deviation!



D. Britzger, J. Currie, A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, C. Gwenlan, A. Huss, T. Morgan, J. Niehuis, J. Pires, K. Rabbertz, M. Sutton, arXiv: 1906.05303.

Scale dependence

Implementation of scale dependence differs between APPLgrid & fastNLO:

APPLgrid:

- Stores one grid per order
- Calculates log terms and convolutions with splitting functions dynamically (HOPPET)
- Computationally more complex, smaller grid sizes

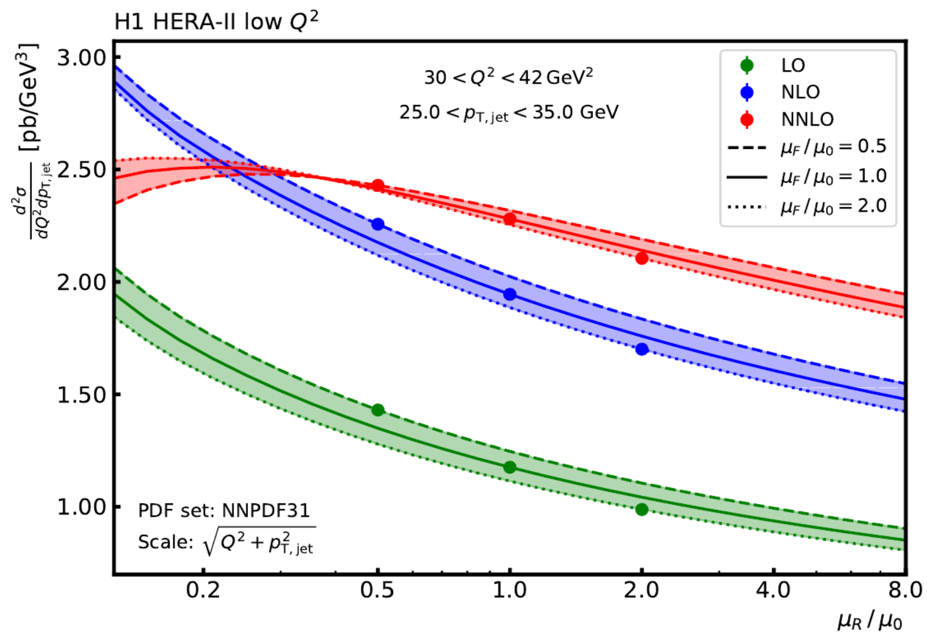
fastNLO:

- Stores one grid per log term per order
- Simpler computation, but larger grid sizes
- Simpler to store alternative scale choice, again increasing grid size

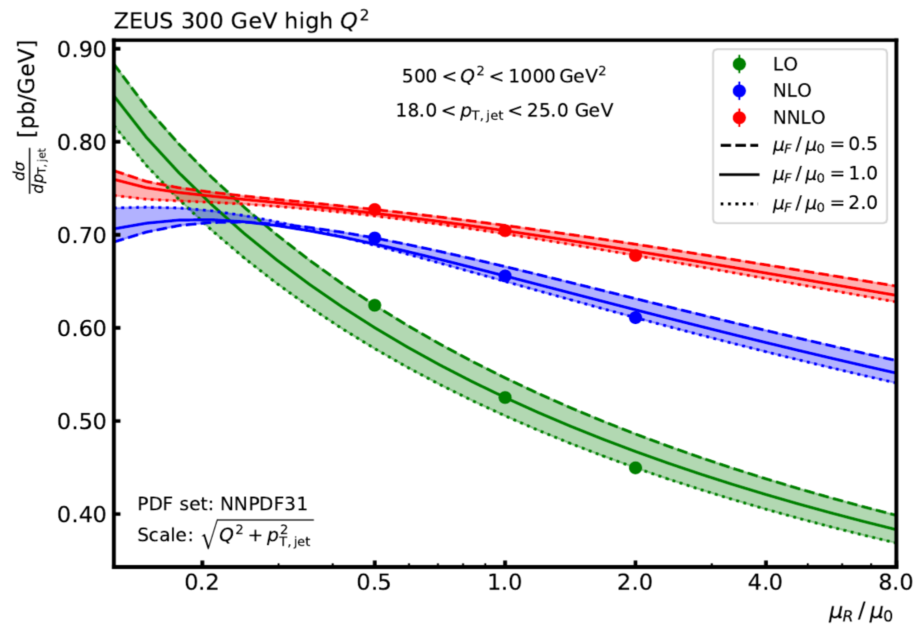
Conversion possible between APPLgrid & fastNLO formats:

- Here: Convert central scale grid from fastNLO production to APPLgrid
- Both implementations of scale dependence agree

H1 low Q^2



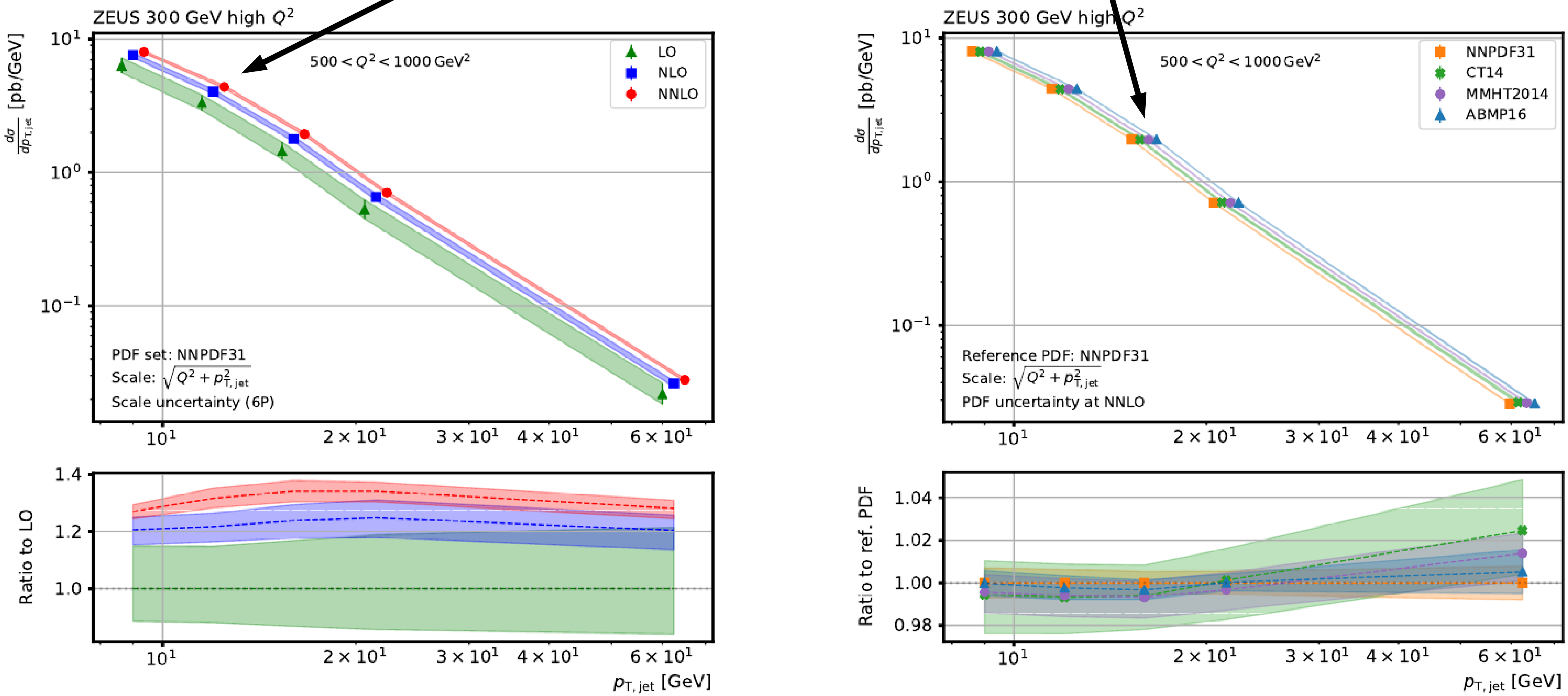
ZEUS high Q^2



Scale & PDF uncertainty

Grids allow fast re-evaluation for different orders, scales, PDFs etc.; here for one of the new ZEUS inclusive jet datasets

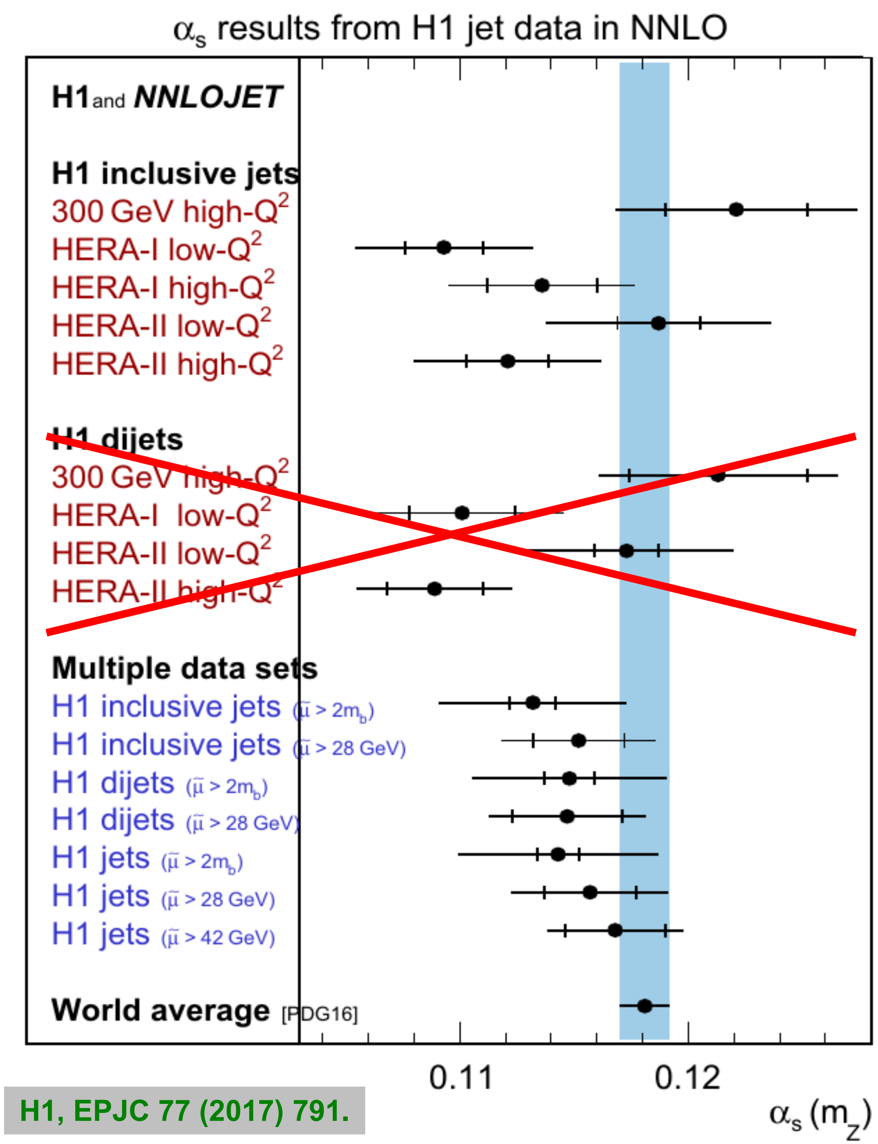
Note: Points are shifted for better visibility



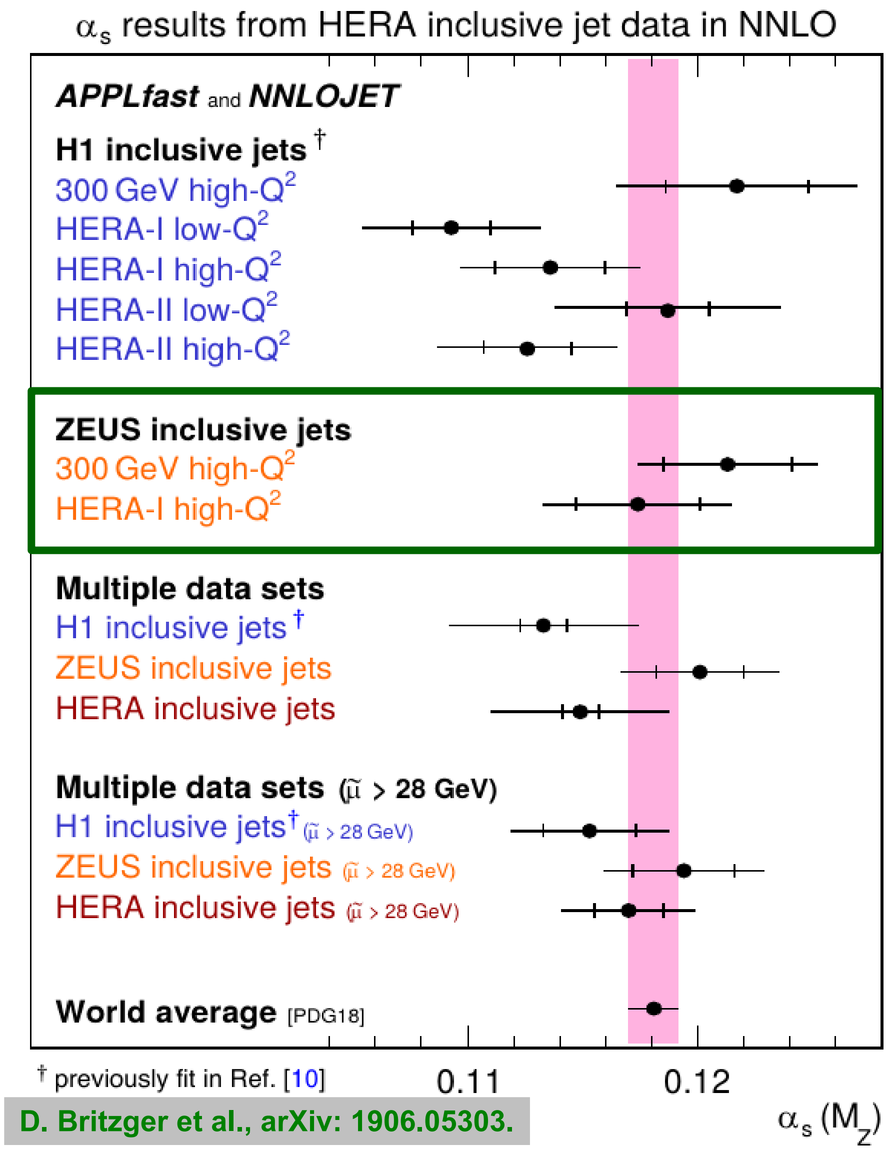
Datasets

Previously:
- H1 data only
- Inclusive & di-jets
- Normalised

Today:
- H1 and ZEUS data
- Inclusive jets only
- Unnormalised
- Some grids with increased stat. precision



H1, EPJC 77 (2017) 791.



D. Britzger et al., arXiv: 1906.05303.

Results

All HERA inclusive jets data points (μ_{cut} > 2 m_b):

$$\alpha_s(M_Z) = 0.1149\,(9)_{\rm exp}\,(5)_{\rm had}\,(4)_{\rm PDF}\,(3)_{{\rm PDF}\alpha_s}\,(2)_{\rm PDFset}\,(37)_{\rm scale}$$

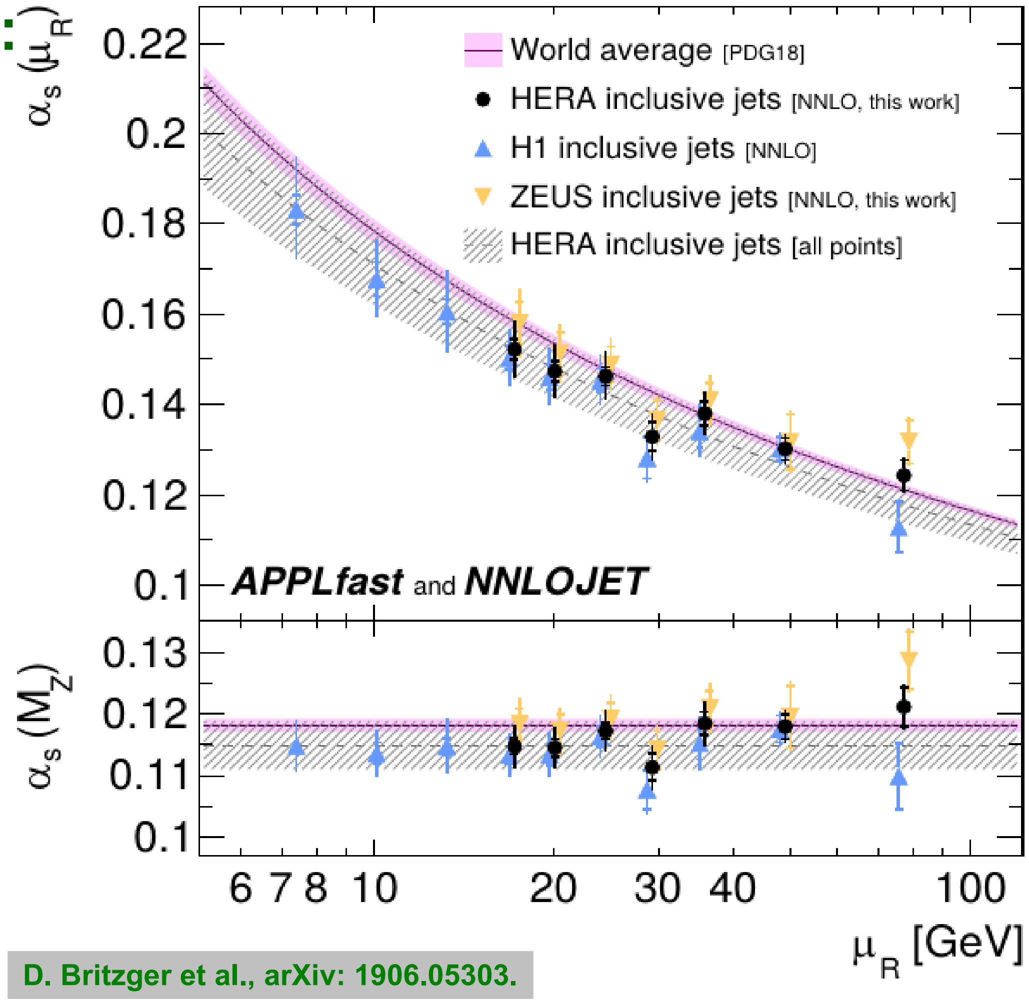
$$\alpha_s(M_Z) = 0.1170\,(15)_{\rm exp}\,(7)_{\rm had}\,(3)_{\rm PDF}\,(2)_{{\rm PDF}\alpha_s}\,(3)_{\rm PDFset}\,(24)_{\rm scale}$$

Smallest total uncertainty (μ_{cut} > 28 GeV):

Compatible with world average

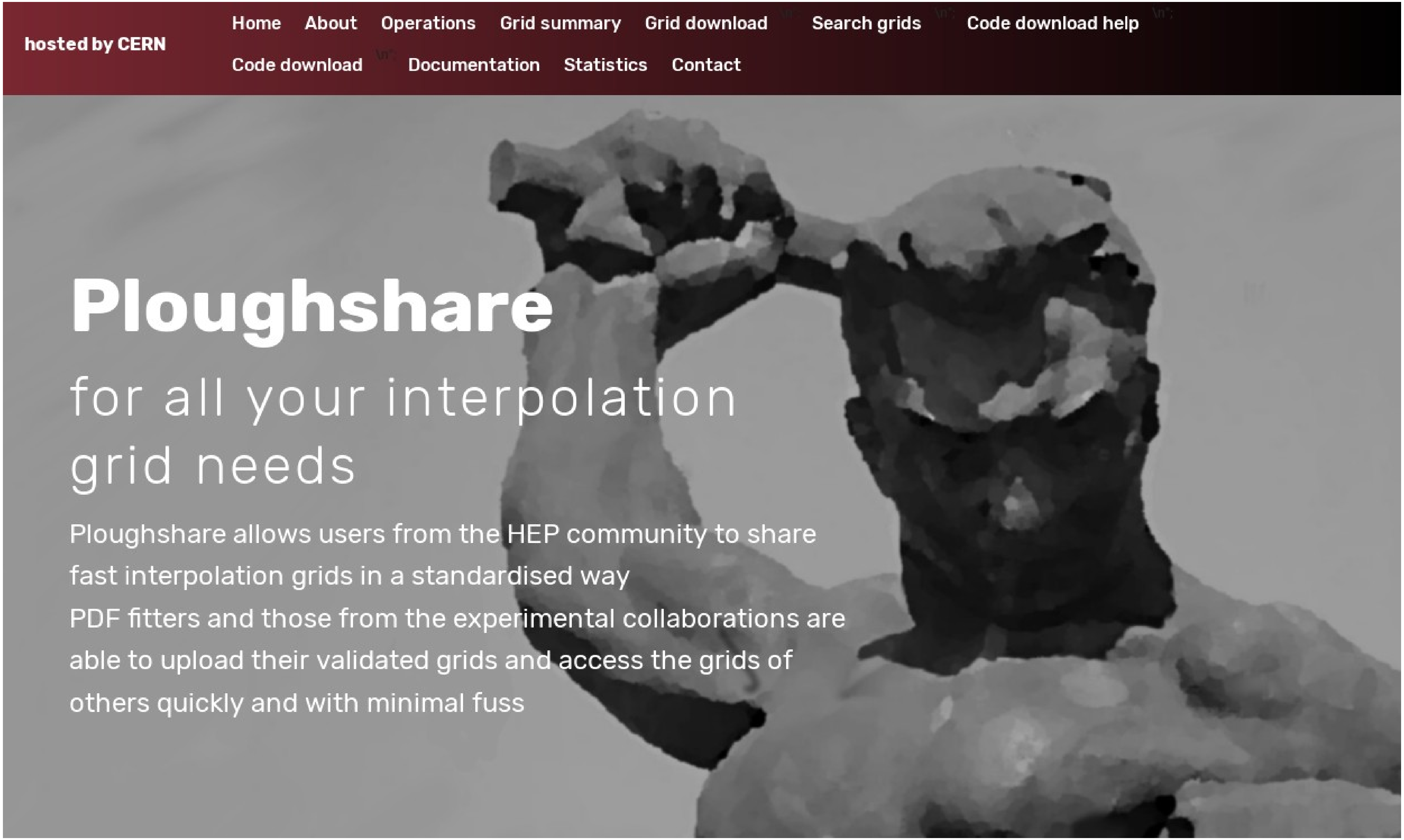
Scale still largest uncertainty :-(

Data	μ_{cut} [GeV]	$\alpha_s(M_Z)$	exp	theo	X2/ ndof
All	2 m_b	0.1149	9	38	183/193
H1	28	0.1153	19	28	44/60
ZEUS	28	0.1194	24	27	39/43
All > 28	28	0.1170	15	25	86/104



D. Britzger et al., arXiv: 1906.05303.

Grid distribution – Ploughshare



Grid distribution – Ploughshare

What is Ploughshare ?

1. **Quick to use** - a web based utility for the automated distribution of fast inperpolation grids for the high energy physcis community.
2. **Secure storage** - registered users can upload grid files and corresponding standard format configuration files to describe the grids and physics processes and these are added to a central repository.
3. **Automatic distribution** - a standard utility library will be provided to download any required grids automatically in user code.

A utility for the community Ploughshare allows users to share their grids, so it is important that the provenance of the grids is guaranteed. This is achieved by allowing only registered users to upload their validated grids. Subsequently however, anyone is free to download and use the grids as they wish.

Fast operations summary

Navigate quickly to some of the primary operations you might be interested in

Download grids
Veiw all the lovely grids which are available for download

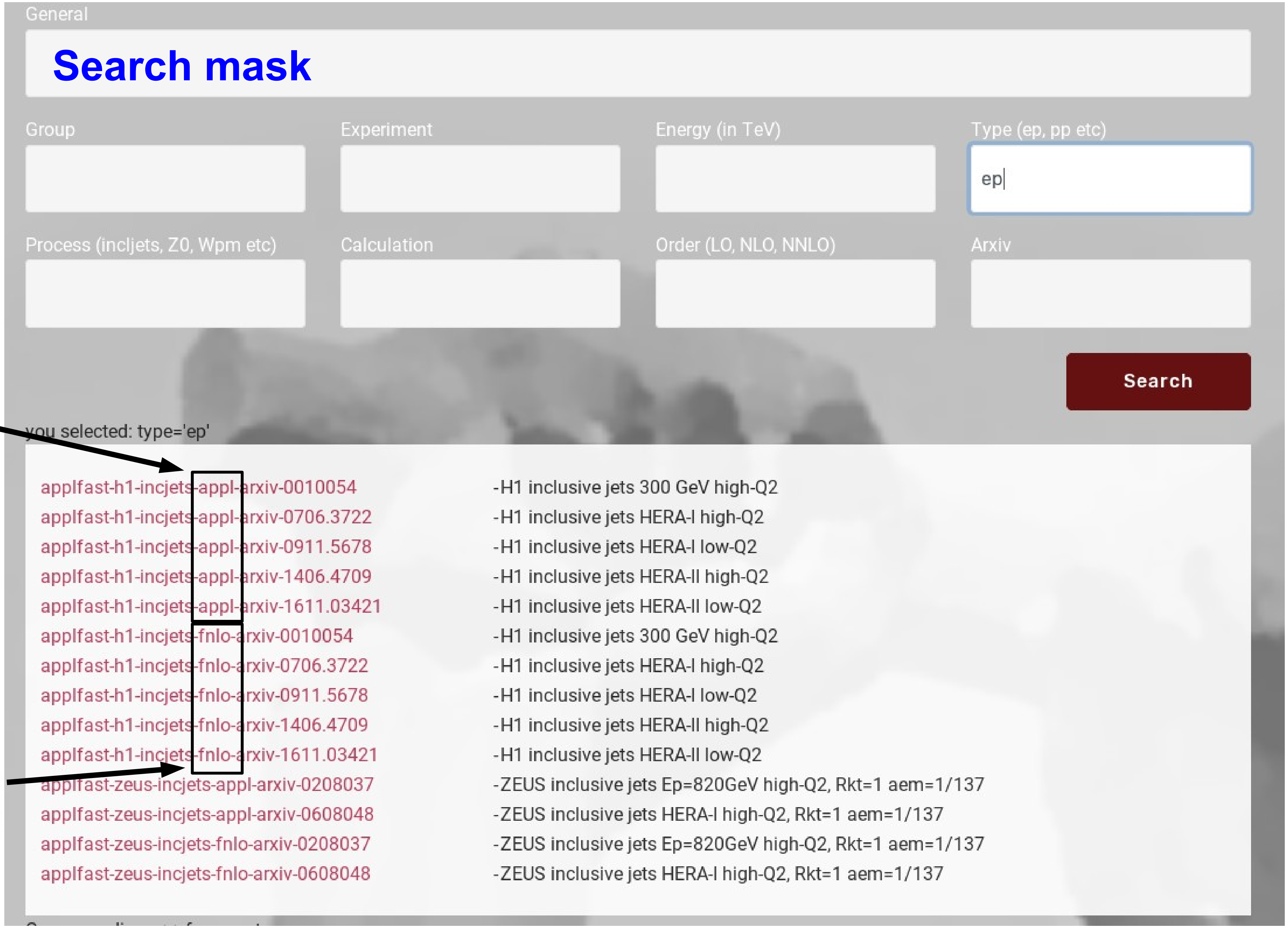
Upload grids
Upload grids using the standard web interface

Download grid code
Get the code for the automated download of multiple grids

Settings
How to set up the automated code for the grid downloads

- **Repository where registered users can upload grids with some documentation**
- **Registered(!) user gets FAME or BLAME**
- **Automated job treats the upload:**
  - **Add to the appropriate location in the file system**
  - **Generate relevant lists, and display web pages**
- **Provides a user interface for automated download with a simple line of code**
- **Have expression of interest from other stakeholders**
- **DIS inclusive jet grids at NNLO are downloadable!**

DIS inclusive jet grids at NNLO



APPLgrid project

Available in both formats

fastNLO

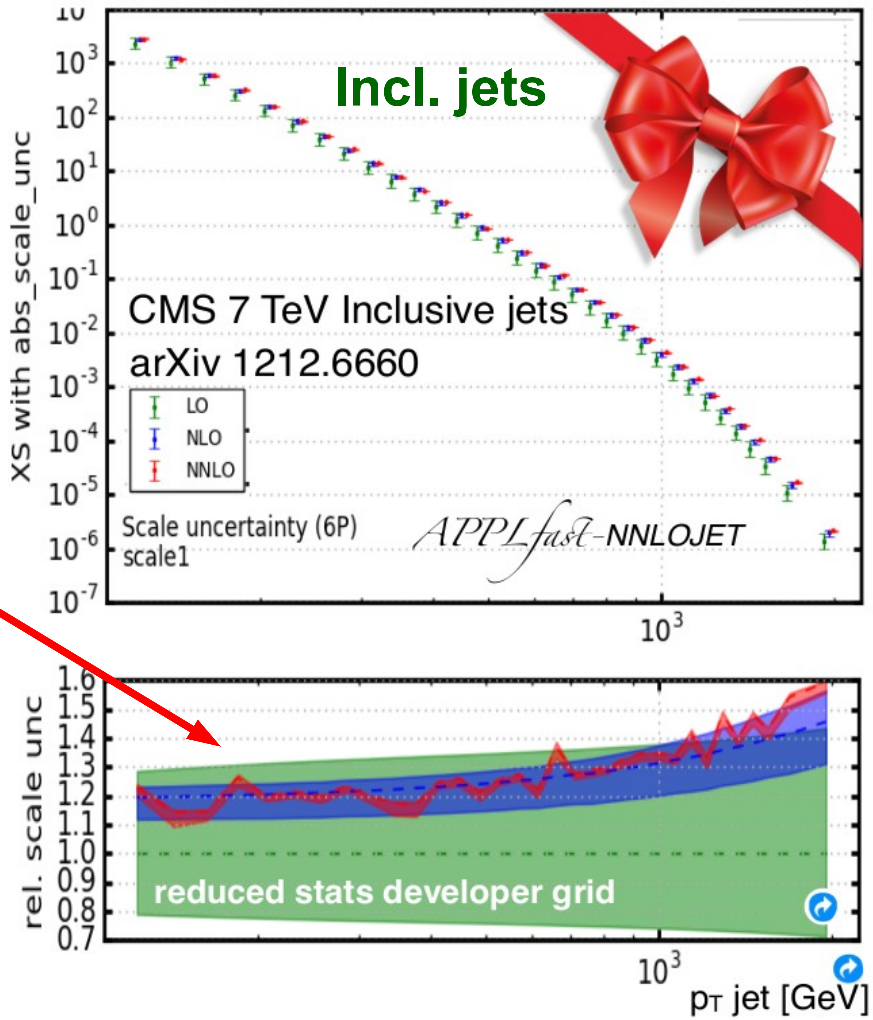
Perspective for pp collisions

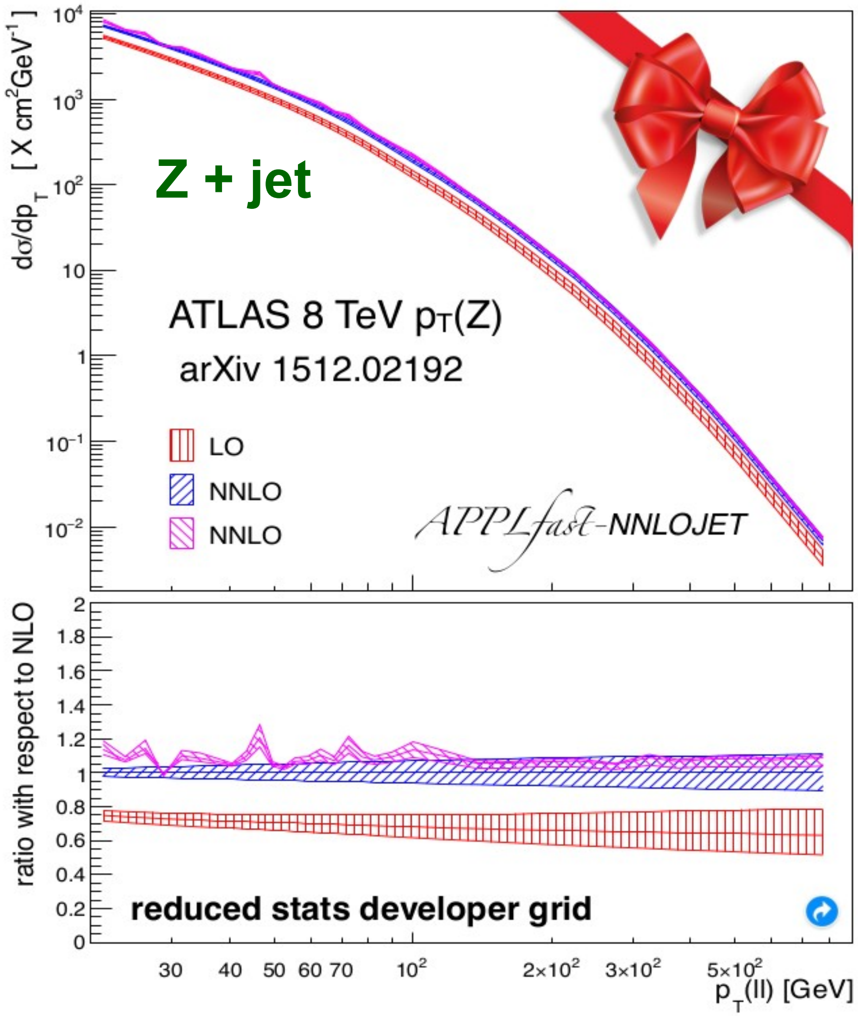
Grid production much more involved: For each event have two x values!

- Grids become substantially larger and need more CPU time for sufficient stat. accuracy
- Nevertheless first grids for testing available

Note: Stat. fluctuations

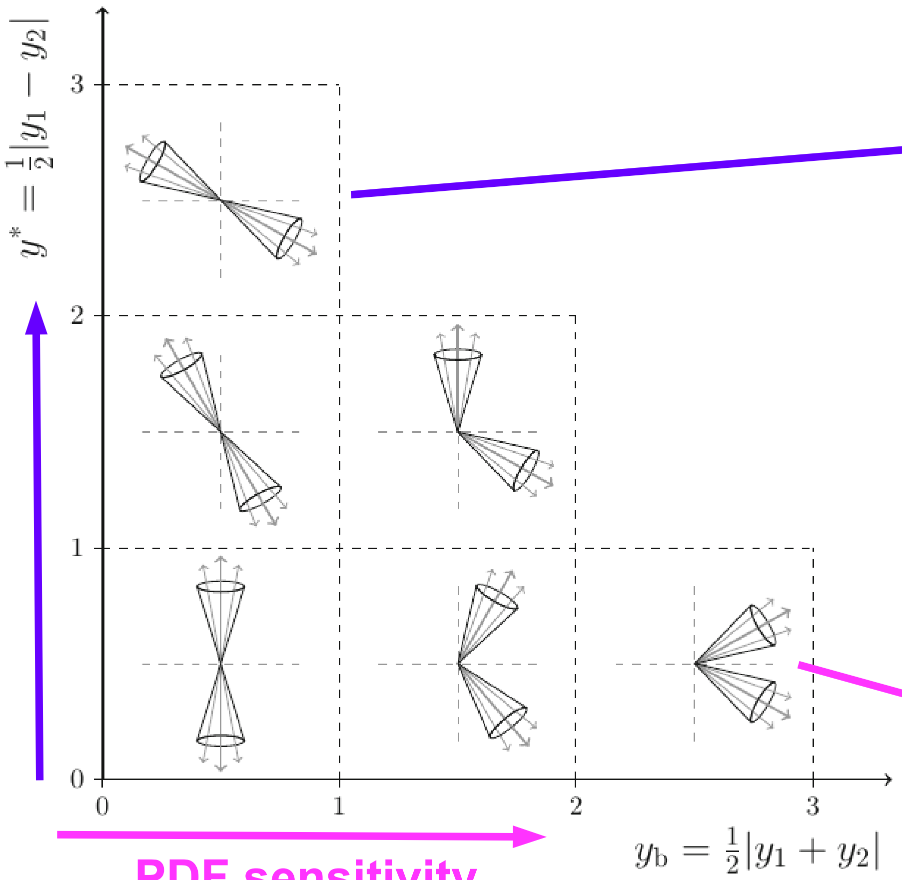
LO, NLO, NNLO scale uncertainty & K factors



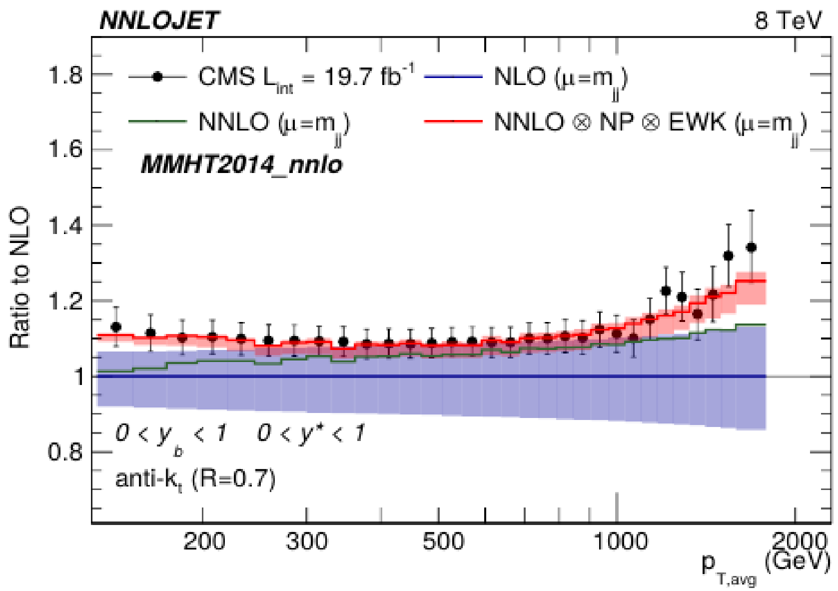


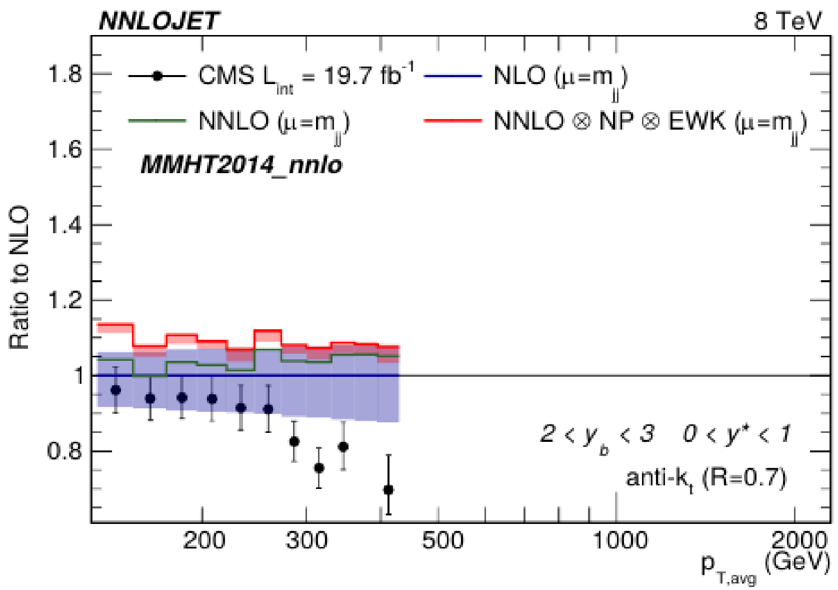
Triple-differential dijet production

Hard process dependent



PDF sensitivity

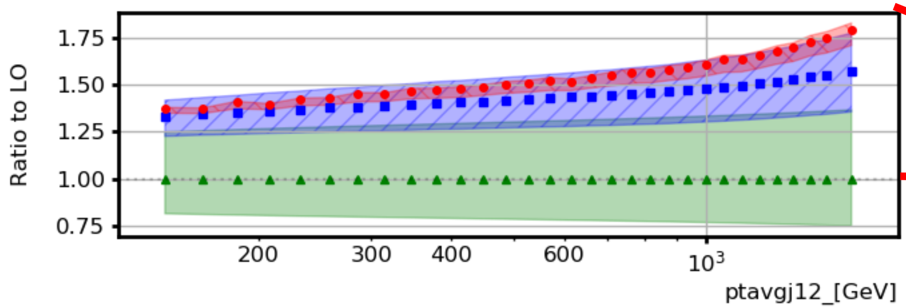




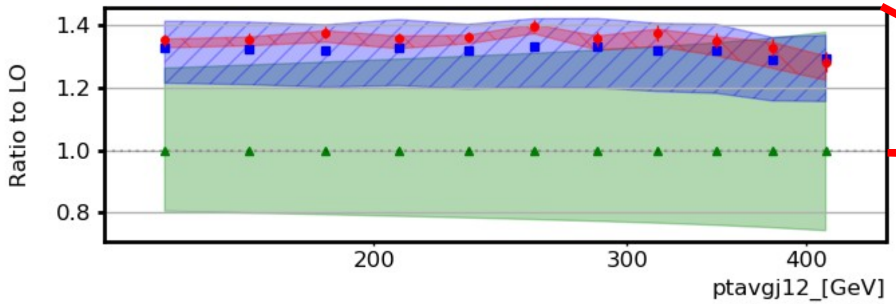
CMS, EPJC 77 (2017) 746, arXiv: 1705.02628.
A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, A. Huss, J. Pires, arXiv: 1905.09047.

Work in progress: Dijets

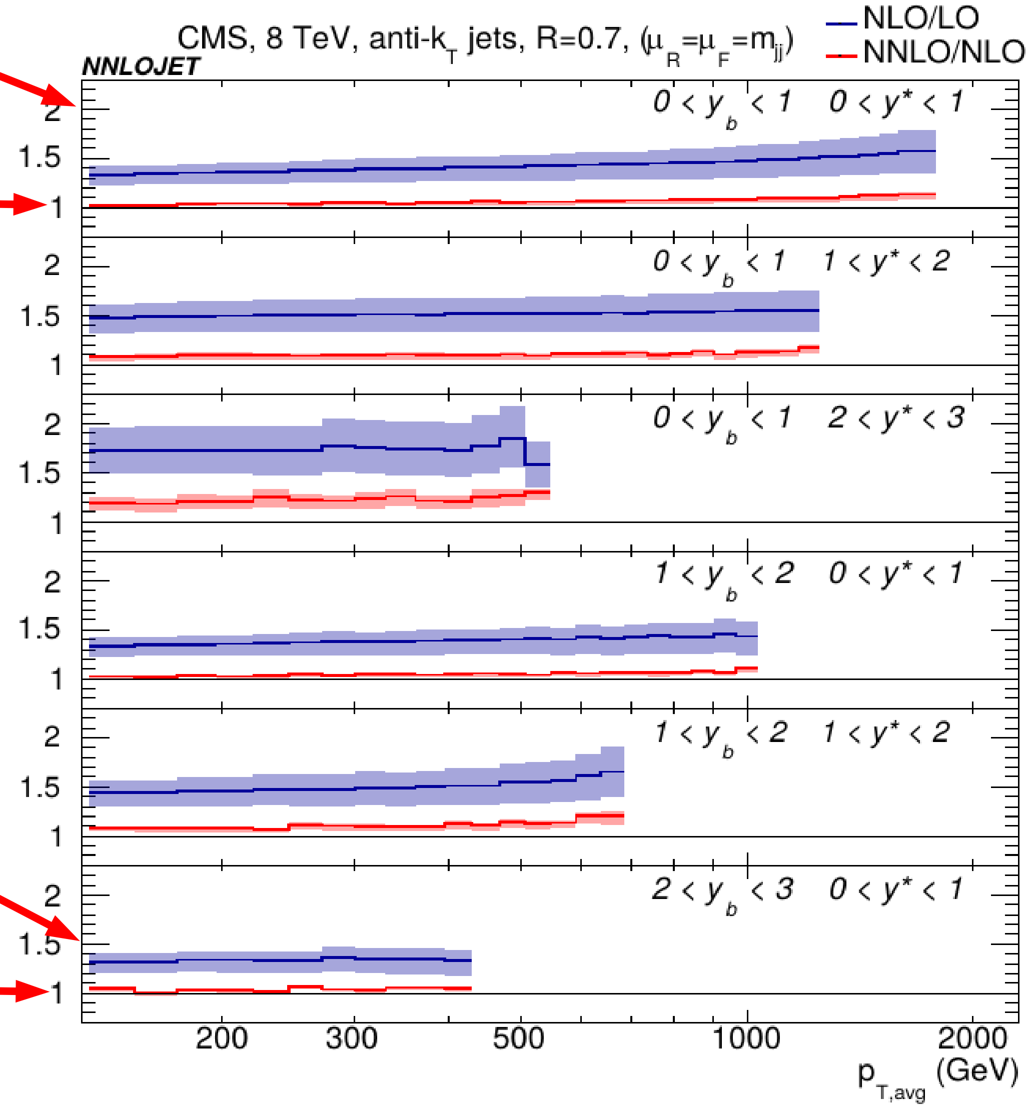
NLO/LO, NNLO/LO



Interpolation table nicely reproduces size and shape of statistically independent NNLOJET original



NLO/LO, NNLO/NLO!



Note: y axis scales differ!

Summary

- APPLfast grids with DIS jets at NNLO from NNLOJET used
  - by H1 to determine α_s from inclusive jets and dijets
  - by HERAPDF team to include jet production in NNLO PDF+α_s fit
  - here to determine α_s from all H1 and ZEUS inclusive jet data
- The best overall uncertainty is achieved for μ_{cut} > 28 GeV

$$\alpha_s(M_Z) = 0.1170 \pm 15(\text{exp}) \pm 25(\text{theo})$$

- Grids are publically available on Ploughshare web site
- Grids for pp processes in progress

Thank you for your attention!

Backup

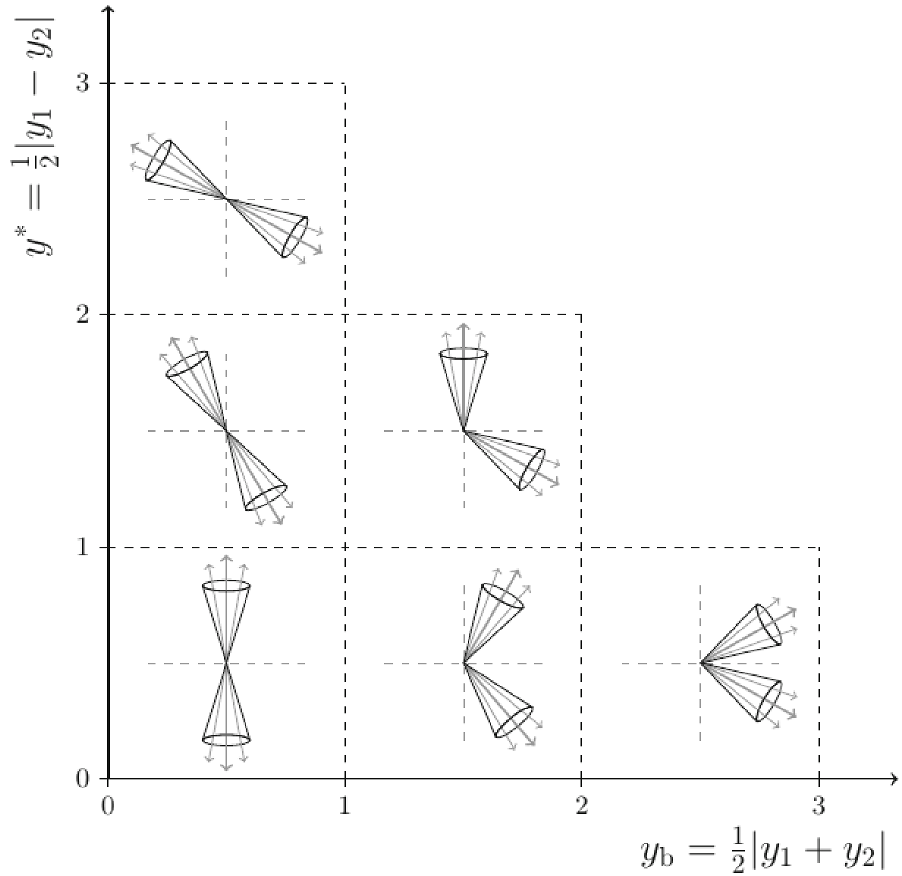
Scale dependence in fastNLO

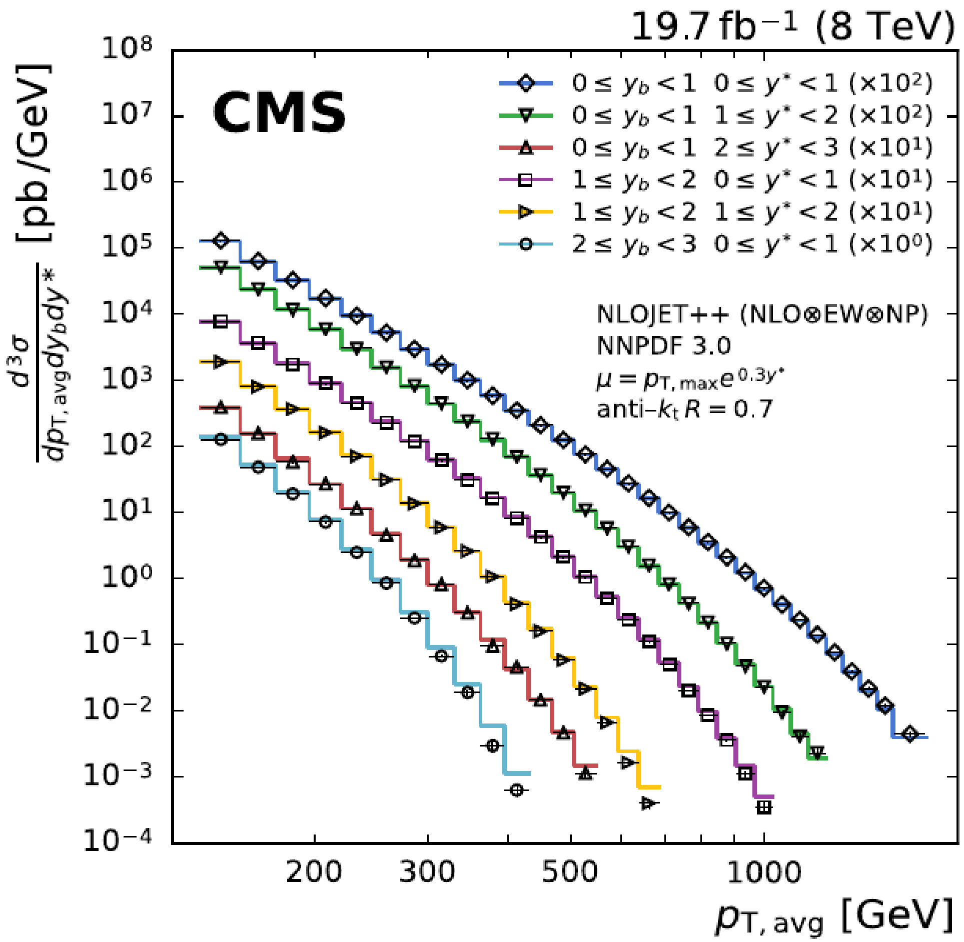
- Storage of scale-independent weights enable full scale flexibility also in NNLO
  - Additional logs in NNLO

$$\omega(\mu_R,\mu_F)=\underbrace{\omega_0+\log(\mu_R^2)\,\omega_R+\log(\mu_F^2)\,\omega_F}_{\text{log's for NLO}}+\underbrace{\log^2(\mu_R^2)\,\omega_{RR}+\log^2(\mu_F^2)\,\omega_{FF}+\log(\mu_R^2)\log(\mu_F^2)\,\omega_{RF}}_{\text{additional log's in NNLO}}$$

  - Store weights: w_0, w_R, w_F, w_{RR}, w_{FF}, w_{RF} for order α_s^{n+2} contributions
- Advantages
  - Renormalization and factorization scale can be varied *independently* and by *any* factor
    - No time-consuming 're-calculation' of splitting functions in NLO necessary
  - Only small increase in amount of stored coefficients
- Implementation
  - *Two* different observables can be used for the scales
    - e.g.: H_T and $p_{T,max}$
    - or e.g.: p_T and $|y|$
    - …
  - *Any function* of those *two observables* can be used for calculating scales

CMS triple-differential dijets





CMS, EPJC 77 (2017) 746, arXiv: 1705.02628.
A. Gehrmann-De Ridder, T. Gehrmann, N. Glover, A. Huss, J. Pires, arXiv: 1905.09047.